CLAIMS

What is claimed is:

1.   A method for generating a schema (22) for a relational database (14) corresponding to a document (12) having a document-type definition (18) and data (16) complying with the document-type definition (18), the document-type definition (18) having content particles representative of the structure of the document data (16), as well as loading the data into the relational database (14) in a manner consistent with the relational schema (22), the method comprising the steps of:

extracting (24) metadata (34) representative of the document-type definition (18) from the document-type definition (18);

generating (28) the schema (22) for the relational database (14) from the metadata (34), wherein at least one table (20) is thereby defined in the relational database (14) corresponding to at least one content particle of the document-type definition (18) via the metadata (34); and

loading (30) the document data (16) into the at least one table (20) of the relational database (14) according to the relational schema (22) in a manner driven by the metadata (34).

2.   The method of claim 1 wherein the extracting (24) step further comprises the step of generating (28) an item metadata table (90) corresponding to element type content particles in the document-type definition (18).

3.   The method of claim 2 wherein the extracting step further comprises the step of creating at least one default item in the item metadata table (90).

4.   The method of claim 3 wherein the extracting step further comprises the step of updating the item metadata table (90) with each of the element type content particles of the document-type definition (18).

-67-

5.   The method of claim 4 wherein the extracting step further comprises the step of generating (28) an attribute metadata table (92) corresponding to attribute type content particles in the document-type definition (18).

6.   The method of claim 5 wherein the extracting step further comprises the step of creating a default attribute value in the attribute metadata table (92) corresponding to attributes of element types in the document-type definition (18).

7.   The method of claim 6 wherein the extracting step further comprises the step of updating the attribute metadata table (92) with each of the attribute type content particles of each element type of the document-type definition (18).

8.   The method of claim 7 wherein the extracting step further comprises the step of generating (28) a nesting metadata table (94) for storing data items corresponding to nesting relationships implied in the document-type definition (18).

9.   The method of claim 8 wherein the extracting step further comprises the step of generating (28) a row in the nesting metadata table (94) corresponding to each relationship between items identified in the item metadata table (90).

10.   The method of claim 9 wherein the generated nesting table (94) row indicates the cardinality between a pair of items.

11.   The method of claim 10 wherein the cardinality is one of one-to-one and one-to-many.

12.   The method of claim 8 wherein the generated nesting table (94) row indicates a relationship between a parent item and a child item.

13. The method of claim 8 wherein the generated nesting table (94) row indicates a relative position of a child item with respect to other items in a definition of the corresponding parent item.

14. The method of claim 7 wherein the generating (28) step further comprises the step of creating at least one table in the schema (22) of the relational database corresponding to at least one row of the metadata item table (90).

15. The method of claim 14 wherein the generating (28) step further comprises generating (28) at least one default field in the table of the schema (22).

16. The method of claim 15 wherein the generating (28) step further comprises the step of altering the schema (22) of the relational database to add at least one column to the at least one table in the relational database (14) schema (22) corresponding to each row of the metadata attribute table (92).

17. The method of claim 16 wherein the generating (28) step further comprises the step of altering the tables in the schema (22) of the relational database to add columns representing links between tables (20) of the relational database schema (22) corresponding to each relationship identified in each row of the metadata nesting table (94).

18. The method of claim 17 wherein the generating (28) step further comprises the step of altering the tables in the schema (22) of the relational database (14) by adding a foreign key to a parent table if the identified relationship is a one-to-one relationship.

19.   The method of claim 18 wherein the generating (28) step further comprises the step of altering the tables in the schema (22) of the relational database (14) by adding a foreign key to a child table if the identified relationship is a one-to-many relationship.

20.   The method of claim 19 and further comprising the step of initializing a link table (36).

21.   The method of claim 19 and further comprising the step of determining whether each item in the metadata nesting table (94) contains a group type.

22.   The method of claim 19 and further comprising the step of initializing a pattern-mapping table (36).

23.   The method of claim 22 and further comprising the step of directly mapping a link into the link table (36) for each item in the metadata nesting table (94) that does not contain a group type.

24.   The method of claim 23 and further comprising the step of creating an additional link table (36) containing a mapping of a link pattern for each group type identified in the metadata item table (90).

25.   The method of claim 24 and further comprising the step of creating a create tuple loading action in the pattern mapping table (36) associated with a particular pattern corresponding to each item in the item metadata table (90).

26.   The method of claim 25 wherein the loading (30) step further comprises the step of creating an update tuple loading action in the pattern mapping table (36) associated with a particular pattern corresponding to each attribute in the attribute metadata table (92).

27. The method of claim 26 wherein the loading step further comprises the steps of:

creating a create tuple loading action in the pattern mapping table (36) associated with a particular pattern corresponding to each group in a link; and

creating an assign action tuple loading action in the pattern mapping table (36) associated with a particular pattern corresponding to each pair in the same link;

corresponding to each link in the link pattern table (36) .

28. The method of claim 27 wherein the loading step further comprises the step of forming a tree structure (300) with the document data (16).

29. The method of claim 28 wherein the loading step further comprises the step of traversing the formed tree (300) and updating the at least one relational database (14) table according to the rows of the pattern mapping table (36).

30. The method of claim 1 and further comprising the step of optimizing (26) the metadata.

31. The method of claim 30 wherein the optimizing (26) step further comprises the step of eliminating duplicate particle references in the metadata (34).

32. The method of claim 31 wherein the optimizing (26) step further comprises the step of simplifying references to corresponding elements, links and attributes in the metadata (34).

33. The method of claim 32 wherein the optimizing (26) step further comprises the step of inlining particular attributes of the metadata (34).

34.   The method of claim 1 wherein the document (12) is an XML document.

35.   The method of claim 1 wherein the document-type definition (18) is a DTD.

36.   The method of claim 1 wherein the data (16) is tagged data.

37.   A system (10) for generating (28) a schema (22) for a relational database (14) corresponding to a document (12) having a document-type definition (18) and data (16) complying with the document-type definition (18), the document-type definition (18) having content particles representative of the structure of the document data (16), as well as loading the data (16) into the relational database (14) in a manner consistent with the relational schema (22), the system comprising:

an extractor (24) adapted to read a document-type definition (18) that extracts metadata (34) representative of the document-type definition (18) from the document-. type definition (18);

a generator (28) operably interconnected to the extractor (24) for generating (28) the schema (22) for the relational database (14) from the metadata (34), wherein at least one table (20) is thereby defined in the relational database (14) corresponding to at least one content particle of the document-type definition (18) via the metadata (34); and

a loader (30) operably interconnected to the generator (28) for loading the document data (16) into the at least one table (20) of the relational database (14) according to the relational schema (22) in a manner driven by the metadata (34).

38.   The system of claim 37 wherein the extractor (24) generates an item metadata table (90) for storing data items corresponding to element type content particles in the document-type definition (18).

39.   The system of claim 38 wherein the extractor (24) creates at least one default item in the item metadata table (90).

40.   The system of claim 39 wherein the extractor (24) generates a row in the item metadata table (90) corresponding to each of the element type content particles of the document-type definition (18).

41.   The system of claim 40 wherein the extractor (24) generates an attribute metadata table (92) corresponding to attribute type content particles in the document-type definition (18).

42.   The system of claim 41 wherein the extractor (24) generates a row in the attribute metadata table (92) corresponding to each of the attribute type content particles of the document-type definition (18).

43.   The system of claim 42 wherein the extractor (24) generates a nesting metadata table (94) for storing data items corresponding to nesting relationship implied in the document-type definition (18).

44.   The system of claim 43 wherein the extractor generates a row in the nesting metadata table (94) corresponding to each relationship identified in the document-type definition (18) between items identified in the item metadata table (90).

45.   The system of claim 44 wherein the generator (28) creates at least one table in the relational database schema (22) of the relational database (14) corresponding to data in the metadata item table (90).

46.   The system of claim 45 wherein the generator (28) alters the schema (22) of the relational database to add a columns to the at least one table of the relational database (14) schema (22) corresponding to each row of the metadata attribute table (92).

47.   The system of claim 46 wherein the generator (28) alters the tables in the schema (22) of the relational database to add columns representing links between tables of the relational database schema (22) corresponding to each relationship identified in each row of the metadata nesting table (94).

48.   The system of claim 47 wherein the generator (28) alters the tables in the schema (22) of the relational database (14) by adding a foreign key to a parent table if a relationship identified between a pair of tables is a one-to-one relationship.

49.   The system of claim 48 wherein the generator (28) alters the tables in the schema (22) of the relational database by adding a foreign key to a child table if a relationship identified between a pair of tables (20) is a one-to-many relationship.

50.   The system of claim 37 and further comprising a link table (36).

51.   The system of claim 50 wherein the system (10) determines whether each item in the metadata nesting table (94) contains a group type content particle.

52.   The system of claim 51 and further comprising a pattern-mapping table (36) in an initialized state.

53.   The system of claim 52 wherein the system (10) directly forms a link in the link table (36) for each item in the metadata nesting table (94) that does not contain a group type.

54.   The system of claim 53 wherein the loader (30) creates an additional link table (36) containing a mapping of a link pattern for each group type identified in the metadata item table (90).

55.   The system of claim 54 wherein the system (10) retrieves a preselected set of rows corresponding to each item in the metadata item table (90).

56.   The system of claim 55 wherein the system (10) creates a create tuple loading action in the pattern mapping table (36) associated with a particular pattern corresponding to each item in the item metadata table (90).

57.   The system of claim 56 wherein the system (10) creates an update tuple loading action in the pattern mapping table (36) associated with a particular pattern corresponding to each attribute in the attribute metadata table (92).

58.   The system of claim 57 wherein the system:

creates a create tuple loading action in the pattern mapping table (36) associated with a particular pattern corresponding to each group in a link; and

creates an assign action tuple loading action in the pattern mapping table (36) associated with a particular pattern corresponding to each pair in the same link;

wherein each created action corresponds to each link in the link pattern table.

59.   The system of claim 58 wherein the loader (30) forms a tree structure (300) with the document data (16).

60.   The system of claim 59 wherein the loader (30) traverses the formed tree structure (300) and updates the at least one relational database (14) table (20) according to the rows of the pattern mapping table (36).

61.   The system of claim 37 and further comprising an optimizer (26) for refining the metadata.

62.   The system of claim 61 wherein the optimizer (26) eliminates duplicate particle references in the metadata (34) .

63.   The system of claim 62 wherein the optimizer (26) simplifies references to corresponding elements, links and attributes in the metadata (34).

64.   The system of claim 37 wherein the document (12) is an XML document.

65.   The system of claim 37 wherein the document-type definition (18) is a DTD.

66.   The system of claim 37 wherein the data (16) is tagged data.

67.   A system (10) for generating (28) a schema (22) for a relational database (14) corresponding to a document (12) having a document-type definition (18) and data (16) complying with the document-type definition (18), the document-type definition (18) having content particles representative of the structure of the document data (16), as well as loading the data (16) into the relational database (14) in a manner consistent with the relational schema (22), the system comprising:

an extractor (24) adapted to read a document-type definition (18) that extracts metadata (34) representative of the document-type definition (18) from the document-type definition (18), wherein the extractor stores the metadata (34) in at least three tables comprising a metadata item table (90) containing metadata (34) representative of element types in the document-type definition (18), a metadata attribute table (92) containing metadata (34) representative of attributes in the document type definition (18), and a

metadata nesting table (94) containing metadata (34) representative of nesting relationships between particles in the document type definition (18).

68.   The system of claim 67 and further comprising a pattern-mapping table (36) initially constructed in an initialized state.

69.   The system of claim 68 wherein the pattern mapping table (36) is loaded with actions indicative of relationships between the data (16) and the document-type definition (18).

70.   The system of claim 67 and further comprising a generator (28) operably interconnected to the extractor (24) for generating (28) the schema (22) for the relational database (14) from the metadata (34), wherein at least one table (20) is thereby defined in the relational database (14) corresponding to at least one content particle of the document-type definition (18) via the metadata (34).

71.   The system of claim 70 wherein the generator (28) forms a table (20) with at least one default field in the relational database (14) for each item contained in the metadata item table (90).

72.   The system of claim 71 wherein the generator (28) forms a column in a corresponding table (20) in the relational schema (22) corresponding to each attribute in the metadata attribute table (92) linked to an item in the metadata item table (90).

73.   The system of claim 72 wherein the generator (28) forms a link between tables in the relational database corresponding to nesting relationships contained in the metadata nesting table (94).

74.   The system of claim 67 and further comprising a loader (30) operably interconnected to the generator (28) for loading the document data (16) into the at least one table (20) of the relational database (14) according to the relational schema (22) and driven by the metadata (34).